CHAPTER 1
INTRODUCTION

1.1 WWW AND INFORMATION RETRIEVAL TOOLS
World Wide Web (WWW) is a huge repository of interlinked hypertext documents known as Web pages. A hypertext document consists of both, the contents and the links to related documents. The links are also known as hyperlinks. Users access these hypertext documents via Internet. A special software known as web browser is used to view the web pages that may contain information in form of text, images, videos and other multimedia and navigate them using hyperlinks known as Uniform Resource Locators (URLs). Though the concept of hypertext is much older but WWW was originated after Tim Berners-Lee, an English physicist wrote a “proposal using hypertext to link and access information in 1990”. Since then websites were created around the world using hypertext mark up languages and connected through Internet. Now it has become an integral part of human life to use Internet to access the information from WWW.

The current population of the world is approximately 6.77 billion out of which approximately 1.67 billion people (24.7%) use Internet [100]. In fact, the number of Internet users has increased from .36 billion in 2000 to 1.67 billion in 2009 i.e. an increase of 362% from 2000 to 2009. The same growth rate is expected in the future too. In Asia alone, around .7 billion people use Internet, that is, approximately 42.2% of worldwide Internet users. As far as India is concerned, it is the third largest country of Internet users in Asia after China and Japan, where approximately .08 billion people use Internet. Thus, it is not too far away when one will start feeling that life is incomplete without Internet.

From the statistics discussed above, one can draw the image of Internet users and so of web documents on WWW, in the coming years. Since its inception in 1990, WWW has become many folds in size and now it contains more than 50 billion publicly accessible web documents [1], distributed all over the world on thousands of web servers and is still growing at an exponential
rate. It is very difficult to search information from such a huge collection of World Wide Web as the web pages/documents are not organized as books are on shelves in a library, nor are web pages completely catalogued at one central location. It is not guaranteed that users will be able to retrieve information even after knowing where to look for information by knowing its URLs as Web is constantly changing and so are the URLs. According to a survey [34], almost 100% of Web users claim to be using search engines or other information retrieval tools to find specific information of interest from WWW. Now people have started talking about Web information retrieval support systems (WIRSSs) instead of just Web information retrieval systems (WIRSs) [107-116]. Under WIRSS approach, the focus changes from the indexing and query matching as used in WIRS to supporting the fundamental knowledge-generating activities of the users. The systems designed for WIRSSs can take advantage of the human visual processing systems aspect to interactively communicate information about the search processes to the users, and thus users can interactively manipulate their search activity features.

All information retrieval tools can be divided into three categories as:

- Web directories
- Meta search engines
- Search engines

### 1.1.1 WEB DIRECTORIES

Web documents are organized in a hierarchical taxonomy tree on the basis of topics and subtopics in Web directories. To access information from a Web directory on a topic, it is necessary to traverse a path in the taxonomy tree from the root to the desired node in the tree. The hierarchical tree is organized in such a way that general topics are sub-divided into more specified topics and so on. Arrangement of the web documents in this hierarchical way helps even a non-expert user to easily access the information. However, the basic problem with a Web directory is that the hierarchical tree is maintained manually and therefore only a small fraction of the Web can be covered. Another problem in the Web directory is that a longer path may be required to be followed to retrieve higher relevant information in comparison to less relevant
information, if it is down the order in the tree [101] whereas this problem is not faced in search engines which use flat approach to access information. So, user can get the most relevant information in one go if an appropriate search query is provided in the beginning. Examples of well known, general, web directories are Yahoo, Looksmart and Open Directory Project (ODP).

1.1.2 META SEARCH ENGINES
Meta search engines are another information retrieval tools that do not maintain their own indexes/repository, rather they are developed to exploit the best features of other Search engines. They provide a single interface, where user queries are provided and these queries are sent to many search engines. Results obtained thereof from multiple search engines are compiled after eliminating duplicate copies and after ranking, the final result is displayed to the users. Example of Meta search engines are MetaCrawler, Dogpile, Copernic, SurfWax, and 37.com.

1.1.3 SEARCH ENGINES
Web directories are not so capable to serve the purpose of surfing the Web consisting of such a huge amount of data. Search engines are another type of information retrieval tools used to access information from the Internet. Starting in 1994, a number of search engines were launched, including AltaVista, Excite, Infoseek, Inktomi, Lycos, and of course the evergreen, Yahoo and Google. Most of these search engines save a copy of the web pages in their central repository and then make appropriate indexes of them for later search/retrieval of information.

Internet would have not become so popular if Search engines would not have been developed. User interface, query engine, indexer, crawler(s) and repository are the major components of a Search engine. To access information from the WWW, users provide search queries in the Search engine’s interface. In response to the search query provided, Search engines use their database to search the relevant documents and produce the result after ranking it on the basis of relevance. Though all components of Search engines play major role, the quality of search results depends on the efficiency of its database/repository. In fact, the Search engine builds its database, with the help of crawlers, where a crawler is a program that traverses the Web and
collects information about web documents. It extracts the links from the collected pages and follows them to download related pages and so on.

1.1.3.1 WEBCRAWLER(S)
All popular search engines maintain their own central repositories of web pages, except Meta search engines. In response to the user query, to facilitate the search from its database, in a fast and effective manner, they create indexes for the repository. So how is it possible to create and manage such a huge index/repository for a Search engine? The crawlers come into picture here. WebCrawler also known as spider, robot, web pot etc is a program that traverses the Web and collects web documents. Web Crawlers start from an initial set of URLs known as seeds, download web document for the seed URLs and extract new links present in these downloaded documents. The extracted URLs are checked, whether web documents, corresponding to them have already been downloaded or not. Once it is ensured that no web documents have been downloaded so far, URLs are reassigned to crawlers for further downloading. The entire process is repeated till no more URLs are left for downloading or target number of web documents has been downloaded. Millions of web pages are downloaded per day by a crawler to achieve the target. There are many types of web crawlers such as parallel crawler, distributed crawlers and focused crawlers.

- PARALLEL WEBCRAWLER
Many search engines run multiple crawlers in parallel to maximize the download rate and to retrieve the whole or significant portion of the Web. These types of crawlers are referred as parallel crawlers. The overlap, quality and network bandwidth are the major issues that make the study of a parallel crawler challenging and interesting [3].

When multiple crawlers run in parallel to download web pages then it may happen that same web pages are downloaded multiple times as one crawler may not be aware that another has already downloaded the page. This problem is known as overlap. By avoiding the overlapping, network bandwidth may be saved and the saved bandwidth may be utilized for further downloading. By doing this it may increase the effectiveness of the crawlers.
The basic aim of any crawler is to download important web pages first but while working in parallel each crawler may not be aware about the whole collection of the web pages downloaded collectively. Their crawling decisions may be based on their local collection of Web and thus may lead towards poor crawling decisions.

In order to minimize overlap and at the same time to maintain the quality of downloaded web pages, individual crawlers need to communicate among themselves to coordinate the whole crawling process. But the drawback with this communication strategy is that it will consume bandwidth as well as the crawler’s time. So, a proper strategy should be applied to minimize the communication while maintaining the quality of crawling.

1.2 MAJOR OBJECTIVES OF SEARCH ENGINES
From above discussions it is very clear that search engines are the most popular information retrieval tools having the following objectives:

1. It should explore and download web documents from WWW as much as possible.
2. It should bring high quality documents so that the user gets the required relevant information within acceptable time.
3. The documents must be displayed in the order of their relevance with respect to the user query.
4. As the web documents are very much dynamic in nature, Search engines should update their repository as frequently as possible. The ideal case would be of synchronizing updation of repository with the web document’s actual change frequency.

To satisfy the first objective i.e. to cover the Web as much as possible, nowadays, Search engines do not depend on a single but on multiple crawlers that execute in parallel, to achieve the target. While working in parallel, crawlers still face many challenging problems such as overlapping, quality, network bandwidth that need to be addressed.
Search engines employ ranking algorithms to meet second and third objectives mentioned above. The most popular algorithm being back link count, proposed by Sergey Brin and Lawrence Page, the Google founders in 1998 [2]. Though back link count helps in efficiently displaying the documents in the order of their relevance but it fails to bring quality documents. The reason being that it needs to have the image of entire Web in terms of back link count over the entire Web, which is not possible for the crawler especially when its database is in the starting or growing stage.

The fourth objective of search engine is to keep its database up-to-date with respect to the web pages maintained at Web server end. The optimum case will be if the updating frequency is synchronized with web page’s change frequency. In fact, it is almost impossible to find the exact change frequencies of web documents as they get changed at random and follow the Poisson process [72]. Nevertheless, it is equally important to find whether a document has changed or not.

1.3 MAIN CONTRIBUTION OF THE THESIS

The focus of this thesis is to investigate the issues in concern to parallel crawling and search engines database updating.

- A survey has been conducted to know the users search trend on WWW. It has helped to identify the major problems users face while searching required information from the Internet.

- A novel architecture for incremental parallel web crawler has been designed that helps to reduce the overlap and network bandwidth problem among crawlers while working in parallel.

- A ranking method has been developed which is used to rank the unvisited URLs and helps to decide their order of visit. This ensures that more relevant documents are visited first enabling the crawlers to download high quality documents.
• Additionally, methods for change detection among web documents have been developed. These methods can be used to decide whether two versions of a web document are same or different and thus help to decide whether the existing web document is required to be replaced or not.

1.4 STRUCTURE OF THE THESIS

The thesis is organized chapter wise as follows:

Chapter 1: This chapter is devoted to introduction about World Wide Web, Internet, Search engines and Web crawlers. It has been discussed as to how the WWW evolved and has become the source of information sought by users. The introduction of various search tools used for searching the information from Internet and challenges faced by them has been discussed.

Chapter 2: In this chapter, a discussion on various searching tools is given. The general architecture of a search engine along with its major components is also given. A detailed discussion of web crawlers and their types is also discussed. Additionally, some popular searching algorithms such as Fish, Shark, Breadth first search, Depth first search, and Best first search have been discussed in this chapter. This chapter also discusses about the existing change detection techniques used to detect whether versions of web documents have been changed or not. Finally, based on the literature reviewed the major challenges being faced by parallel web crawler and their change detection techniques have been identified, providing the basis for the work to be carried out.

Chapter 3: This chapter deals in detail about a survey which was conducted to identify the users search behavior on WWW while using search engines. The various problems faced by them were identified. Based on the responses received and the analysis carried thereof, inferences were drawn.

Chapter 4: In this chapter a novel architecture for incremental parallel web crawler has been proposed. The proposed architecture not only minimizes the overlap problem but also avoids unnecessary communications among client crawlers along with maintaining quality of
collectively downloaded web pages. The details about the various components and their working have been provided. While concluding the chapter, a comparison between the proposed and existing architecture of parallel crawlers has been made.

**Chapter 5:** In this chapter, mechanisms that determine whether a web document has been changed over time from its previous version or not and by what amount have been proposed. The hallmark of the proposed mechanisms is that changes occurring at micro level i.e. paragraph levels are also identified with a capability to identify three major types of changes namely: content level changes, structural changes and presentation changes. At the end of this chapter the performance of the proposed methods with some well known change detection algorithms have been compared.

**Chapter 6:** It is the last chapter of the thesis in which conclusion and future work have been discussed. It has been concluded that the architecture of incremental parallel WebCrawler, designed in this work, has achieved the objectives identified during the literature survey.

The change detection methods for Structural, Presentation and Content level changes have been implemented and their efficacy has been tested. In the end, the publications which have been referred during the work have been listed followed by two appendices. Appendix A provides the details of implementation of parallel crawler and appendix B contains the web pages which were used to test the efficiency of change detection methods.

The issues of identifying changes in an image embedded, and detection of behavioral changes in web documents have been suggested for the work to be carried out in future.